

[Previous Doc](#)

[Next Doc](#)

[Go to Doc#](#)

[First Hit](#)



Generate Collection

L2: Entry 26 of 26

File: JPAB

Aug 19, 1997

DOCUMENT-IDENTIFIER: JP 09216349 A

TITLE: DATA PROCESSING METHOD IN PAGE PRINTER

Abstract Text (1):

PROBLEM TO BE SOLVED: To obtain a low costed high resolution page printer by reducing a memory capacity of a frame buffer by a method wherein a variable reversible compression process is applied to each size of a data which is rasterized again, and a compression code is stored in a compression memory.

[Previous Doc](#)

[Next Doc](#)

[Go to Doc#](#)

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平9-216349

(43) 公開日 平成9年(1997)8月19日

(51) Int.Cl. <sup>6</sup>	識別記号	庁内整理番号	F I	技術表示箇所
B 4 1 J 2/00			B 4 1 J 3/00	Z
				5/30 Z
G 0 6 F 3/12			G 0 6 F 3/12	B
G 0 6 T 1/00			H 0 4 N 1/41	Z
// H 0 4 N 1/41			G 0 6 F 15/66	J

審査請求 未請求 請求項の数 5 F D (全 18 頁)

(21) 出願番号 特願平7-354250

(22) 出願日 平成7年(1995)12月29日

(71) 出願人 000006747

株式会社リコー

東京都大田区中馬込1丁目3番6号

(72) 発明者 柳下 高弘

東京都大田区中馬込1丁目3番6号 株式会社リコー内

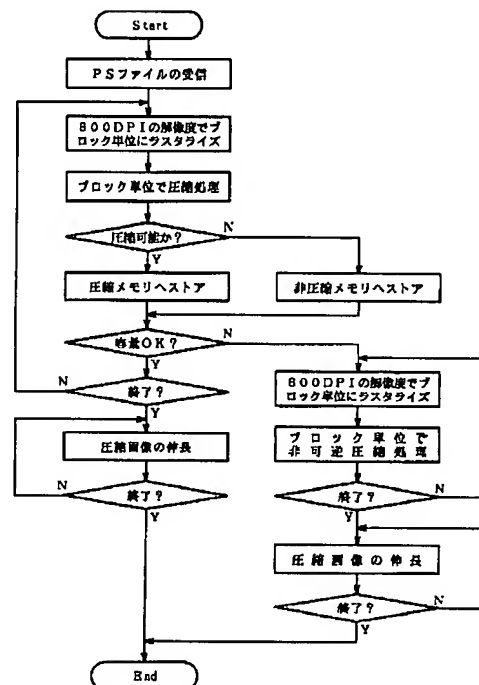
(74) 代理人 弁理士 大澤 敬

(54) 【発明の名称】 ページプリンタにおけるデータ処理方法

(57) 【要約】

【課題】 ページプリンタにおけるフレームバッファのメモリ容量を大幅に削減し、低価格で高解像度のページプリンタを提供する。

【解決手段】 印刷すべき文書データを通常印刷で用いられる600DPIの解像度でラスタライズし、それをブロック単位で可変長可逆圧縮処理を施して、その圧縮コードを圧縮メモリに格納する。しかし、1ページ分の処理が終了しないうちに圧縮コードがメモリ容量を越えた場合には、600DPIの解像度で再度ラスタライズし直し、それをブロック単位で可変長非可逆圧縮処理を施して、その圧縮コードを圧縮メモリに格納する。その後これらの圧縮コードを順次伸長処理してビデオ出力をプリンタエンジンへ送出して印刷させる。



## 【特許請求の範囲】

【請求項1】 ページプリンタにおいて、印刷すべき文書データを通常印刷で用いられる解像度にラスターライズし、そのラスターライズしたデータに対してある大きさ毎に可変長可逆圧縮処理を施し、その結果としての圧縮コードを圧縮メモリに蓄えると共に、

前記圧縮コードが所定の大きさを越えた場合には、印刷すべき文書データを前記解像度でラスターライズし直し、そのラスターライズし直したデータに対してある大きさ毎に可変長可逆圧縮処理を施し、その結果としての圧縮コードを前記圧縮メモリに蓄えることを特徴とするデータ処理方法。

【請求項2】 ページプリンタにおいて、印刷すべき文書データを第1の解像度でラスターライズし、そのラスターライズしたデータに対してある大きさ毎に可変長可逆圧縮処理を施し、その結果としての圧縮コードを圧縮メモリに蓄えると共に、

前記ラスターライズを開始してから前記圧縮コードが所定の大きさを越えるまでの処理時間を計測し、

前記圧縮コードが前記所定の大きさを越えたとき、計測した処理時間が所定値より小さかった場合は、前記第1の解像度より小さい第2の解像度でラスターライズし直し、前記計測した処理時間が所定値より大きかった場合は、前記第1の解像度でラスターライズされたデータを前記第2の解像度に変換し、その後該第2の解像度でラスターライズを継続することを特徴とするデータ処理方法。

【請求項3】 ページプリンタにおいて、印刷すべき文書データを、そのページ言語中に含まれる印字範囲を示すデータが所定の値より小さい場合は第1の解像度でラスターライズし、そのラスターライズしたデータに対してある大きさ毎に可変長可逆圧縮処理を施し、前記印字範囲を示すデータが所定の値より大きい場合は、前記第1の解像度より小さい第2の解像度でラスターライズすることを特徴とするデータ処理方法。

【請求項4】 画像品質を優先するモードと処理速度を優先するモードとが選択可能なページプリンタにおいて、

前記画像品質を優先するモードが選択された場合は、印刷すべき文書データを通常印刷で用いられる第1の解像度でラスターライズし、そのラスターライズしたデータに対してある大きさ毎に可変長可逆圧縮処理を施し、前記処理速度を優先するモードが選択された場合は、前記印刷すべき文書データを前記第1の解像度よりも小さい第2の解像度でラスターライズすることを特徴とするデータ処理方法。

【請求項5】 請求項1乃至4のいずれか一項に記載のデータ処理方法において、前記ラスターライズされたデータに対する可変長可逆圧縮処理を、

主走査方向にもドットで構成されるブロックと、主走査

方向あるいは副走査方向にuブロックで構成されるユニットの2つの基本単位を用い、印刷すべきデータを1ページ分ずつラスターライズするために必要なページバッファを前記ユニット単位に分割するステップと、

その分割した各ユニットのデータが全白であるかどうかをチェックし、その結果をmビットのホワイトマップテーブルに保存するステップと、

前記ユニットのデータが全白でない場合に、前記ホワイトマップテーブルで指定されるアドレス領域にuブロック分のnビットの圧縮データテーブルを用意し、u個分のブロック毎のデータの圧縮処理結果を格納するステップと、

前記ブロック毎の圧縮処理が不可能な場合に、前記圧縮データテーブルで指定されるアドレス領域にもビットの非圧縮データテーブルを用意し、該ブロックのデータをそのまま格納するステップと、

によって行なうことを特徴とするデータ処理方法。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】この発明は、白黒およびカラーのレーザプリンタ、LEDプリンタ等のページプリンタ（デジタル複写機のプリンタ部も含む）において、そのページメモリのメモリ容量を削減し、且つ高画質化を図るためのデータ処理方法に関する。

【0002】

【従来の技術】近年、レーザプリンタ等のページプリンタの解像度は益々高くなり、最近では600DPIが主流になっている。今後は800DPIや1200DPIのプリンタが出現するであろうし、マルチ・リゾリューション・プリンタ、すなわちエミュレーションに応じてエンジンの解像度が変化するプリンタも増えてくるであろう。

【0003】しかし、ページプリンタ内の1ページ分のビットマップデータを展開するフレームバッファのメモリ容量は、A4、600DPIで約4MB、A3、600DPIでは約8MBになる。このようなフレームバッファのメモリ容量の増加は、製品価格に大きな影響を与える。一方、ページプリンタの価格は逆に低価格へと推移しているため、高解像度化に伴う価格上昇をなんらかの手段で抑えなければならない。

【0004】そのための2値画像データの可逆圧縮技術には、ファクシミリ等で用いられるMH（ハフマン符号化方式：一次元圧縮法によるG3ファクシミリの標準符号化方式）、MR（二次元圧縮法によるG3ファクシミリではオプションの符号化方式）、MMR（MR符号化方式の変形で、G4ファクシミリの標準符号化方式）や、コンピュータで扱うテキストファイルやバイナリファイルに用いられる、例えばLZ方式の圧縮等がある。また、最近では国際標準である算術符号化を用いたJBIG方式もある。

## 【0005】

【発明が解決しようとする課題】しかしながら、これらの2値画像データ圧縮技術は、画像あるいはファイルの先頭から順番に逐次符号化処理を行なうので、所望の任意ブロックだけをリアルタイムに復元することはできない。

【0006】ページプリンタでは、印刷すべき文書情報がコンピュータのCRT上で実際に作成された順番にくるため、ある大きさのブロック単位で圧縮処理を施さなければならない。従って、上記圧縮技術をそのまま使用することはできない。また、2値画像データの固定長可逆圧縮技術は世の中に存在しない。従って、対象とする画像データによっては、所望の容量に圧縮できない場合がある。

【0007】この発明は上述のような現状に鑑みてなされたものであり、ページプリンタにおけるフレームバッファのメモリ容量を圧縮技術を用いて削減し、低価格な高解像度ページプリンタを提供できるようにすることを第1の目的とする。また、高解像度での圧縮データがメモリ容量を越えるような場合でも、印刷不能になるようなことなく、若干画質を落しても印刷可能にすることも目的とする。

## 【0008】

【課題を解決するための手段】この発明は上記の目的を達成するため、ページプリンタにおいて、次のような各種のデータ処理方法を提供するものである。印刷すべき文書データを通常印刷で用いられる解像度にラスターライズし、そのラスターライズしたデータに対してある大きさ毎に可変長可逆圧縮処理を施し、その結果としての圧縮コードを圧縮メモリに蓄える。そして、上記圧縮コードが所定の大きさを越えた場合には、印刷すべき文書データを上記解像度でラスターライズし直し、そのラスターライズし直したデータに対してある大きさ毎に可変長非可逆圧縮処理を施し、その結果としての圧縮コードを圧縮メモリに蓄える。

【0009】また、印刷すべき文書データを第1の解像度でラスターライズし、そのラスターライズしたデータに対してある大きさ毎に可変長可逆圧縮処理を施し、その結果としての圧縮コードを圧縮メモリに蓄える。さらに、そのラスターライズを開始してから上記圧縮コードが所定の大きさを越えるまでの処理時間を計測する。

【0010】そして、上記圧縮コードが上記所定の大きさを越えたとき、計測した処理時間が所定値より小さかった場合は、上記第1の解像度より小さい第2の解像度でラスターライズし直し、上記計測した処理時間が所定値より大きかった場合は、上記第1の解像度でラスターライズされたデータを第2の解像度に変換し、その後第2の解像度でラスターライズを継続するようにしてもよい。

【0011】さらに、印刷すべき文書データを、そのページ言語中に含まれる印字範囲を示すデータが所定の値

より小さい場合は第1の解像度でラスターライズし、そのラスターライズしたデータに対してある大きさ毎に可変長可逆圧縮処理を施す。一方、上記印字範囲を示すデータが所定の値より大きい場合は、上記第1の解像度より小さい第2の解像度でラスターライズするようにしてもよい。

【0012】あるいは、画像品質を優先するモードと処理速度を優先するモードとが選択可能なページプリンタにおいて、画像品質を優先するモードが選択された場合は、印刷すべき文書データを通常印刷で用いられる第1の解像度でラスターライズし、そのラスターライズしたデータに対してある大きさ毎に可変長可逆圧縮処理を施し、処理速度を優先するモードが選択された場合は、上記印刷すべき文書データを上記第1の解像度よりも小さい第2の解像度でラスターライズするようにすることもできる。

【0013】そして、これらの画像データ処理方法における上記ラスターライズされたデータに対する可変長可逆圧縮処理を、次の各ステップによって行なうことができる。

(1) 主走査方向にもドットで構成されるブロックと、主走査方向あるいは副走査方向にuブロックで構成されるユニットの2つの基本単位を用い、印刷すべきデータを1ページずつラスターライズするために必要なページバッファを前記ユニット単位に分割するステップ、

【0014】(2) その分割した各ユニットのデータが全白であるかどうかをチェックし、その結果をmビットのホワイトマップテーブルに保存するステップ、(3) 上記ユニットのデータが全白でない場合に、上記ホワイトマップテーブルで指定されるアドレス領域にuブロック分のnビットの圧縮データテーブルを用意し、u個分のブロック毎のデータの圧縮処理結果を格納するステップ、(4) 上記ブロック毎の圧縮処理が不可能な場合に、上記圧縮データテーブルで指定されるアドレス領域にもビットの非圧縮データテーブルを用意し、そのブロックのデータをそのまま格納するステップ、

## 【0015】

【発明の実施の形態】以下、この発明の実施の形態を図面に基づいて具体的に説明する。図2はこの発明によるデータ処理方法を実施したページプリンタを用いた画像形成システムの構成例を示す図であり、100はパーソナルコンピュータ、200がページプリンタである。通常、ユーザはパーソナルコンピュータ100のCRT画面とキーボードとDTP（デスクトップ・パブリッシング）アプリケーションを使って文書を作成し、プリンタドライバ101を通してページプリンタ200に転送して印刷を行なう。

【0016】プリンタドライバ101は、通常印刷すべき文書内容をページプリンタ200がサポートするプリンタ言語、例えばヒューレット・パッカード（HP）社

のPCLやAdobe社のポストスクリプト言語等にコンバートする。前者に接続されたプリンタを一般にPCLエミュレーション・プリンタ、後者に接続されたプリンタをポストスクリプト・プリンタと呼ぶ。また、パーソナルコンピュータ100側で全てラスライズされたビットイメージを印刷するだけのダムプリンタと呼ばれるプリンタもある。

【0017】ページプリンタ200としては、レーザプリンタやインクジェットプリンタ、あるいはサーマルプリンタ等があるが、高速印刷という点でレーザプリンタが勝る。最近ではカラーのレーザプリンタも市場に出始めており、それを使用することも可能である。また、解像度は年々高まる一方で、現在は600DPIが標準となっている。以下の実施例では、ページプリンタ200を600DPIの白黒ポストスクリプト・レーザプリンタ（以下単に「ページプリンタ」と称する）として説明するが、この発明はそれに限定されるものではない。

【0018】図3はそのページプリンタの外観図であり、図4はその内部機構の概略を示す縦断面図である。このページプリンタ200は、給紙トレイ2を着脱可能に備え、上部に第1排紙スタッカ3を設け、後部に第2排紙スタッカ4を設けている。2個の排紙スタッカ3、4への排紙は切換爪5によって切換え可能である。通常は、排紙スタッカとして第1排紙スタッカ3が選択されるが、封筒や葉書などのカールし易い紙を使用する場合など、特別な場合に第2排紙スタッカ4が選択される。

【0019】さらに、内部にはプリンタエンジンの作像部を構成する感光体ドラム10、帯電部11、光書込部12、現像部13、転写部14、定着部15と、給紙ローラ16及びレジストローラ対17等による給紙部と、搬送ローラとペーパーガイド板等からなる排紙用搬送部18と、このページプリンタ全体を制御するプリンタコントローラを構成するコントローラ基板19及びプリンタエンジンのシーケンスコントローラを構成するエンジンドライバ基板20等が設けられている。

【0020】そして、プリンタエンジンのシーケンスコントローラによりプリントシーケンスが開始されると、給紙ローラ16によって給紙トレイ2から給紙を始め、その用紙の先端をレジストローラ対17に突き当てた状態で一時停止させる。一方、感光体ドラム10は図4の矢印A方向へ回転し、帯電部11で帯電された表面に、光書込部12によってプリンタコントローラからの画像データに応じて変調されたレーザビームを、ドラム軸方向に主走査しながら照射して露光し、感光体ドラム10の表面に静電潜像を形成する。

【0021】それを現像部13でトナーによって現像し、レジストローラ対17によって所定のタイミングで給送される用紙に転写部14において転写し、定着部15で加熱定着したプリント紙を第2排紙スタッカ4へ送出するか、排紙用搬送部18を通して上部の第1排紙ス

タッカ3へ搬送する。

【0022】図5はコントローラ基板19の内部ブロック図である。このコントローラ基板19は、CPU201、NVRAM203、プログラムROM204、フォントROM205、RAM206、及び4個のインタフェース（以下「I/F」と略称する）207、209、211、213と、これらを接続するバスライン215によって構成されている。

【0023】CPU201は、プログラムROM204に格納されたプログラム、操作パネル210からのモード指示、ホスト装置であるパーソナルコンピュータ（パソコン）100からのコマンド等によって、このコントローラ全体を制御する。また、挿着されたICカード202から、フォントデータやプログラム等を取り込むこともできる。NVRAM203は、操作パネル210からのモード指示の内容などを記憶しておく不揮発性メモリである。

【0024】プログラムROM204は、このコントローラの制御プログラムを格納している読出し専用メモリである。フォントROM205は、文字フォントのパターンデータなどを記憶する。RAM206は、CPU201のワークメモリ、入力データのインプットバッファ、プリントデータのページメモリ（フレームバッファ）、ダウンロードフォント用のメモリ等に使用するランダムアクセス・メモリである。

【0025】エンジンI/F207は、実際に印刷を行なうプリンタエンジン208と接続されて、コマンド及びステータスや印字データの通信を行なうインタフェースである。パネルI/F209は、操作パネル210と接続されて、コマンド及びステータスの通信を行なうインタフェースであり、操作パネル210は、使用者に現在のプリンタの状態を表示して知らせたり、使用者がモード指示を行なったりするパネル装置である。

【0026】ホストI/F211は、ホスト装置であるパーソナルコンピュータ100と通信を行なうインタフェースであり、通常はセントロニクスI/FやRS232Cを使用する。ディスクI/F213は、ディスク装置214と通信を行なうためのディスクインタフェースである。ディスク装置214は、フォントデータやプログラム、あるいは印字データなどの種々のデータを記憶しておくための外部記憶装置であり、フロッピーディスク装置やハードディスク装置などである。

【0027】次に、このページプリンタ200によるページ印刷の動作について説明する。図1は、図5に示したコントローラ基板19及び600DPIのプリンタエンジン208による、ページ印刷の処理に係わる部分の第1の実施形態の機能構成を示すブロック図である。そのPSインタープリタ21と圧縮部22及び伸長部25は、図5のCPU201及び後述するASIC（特定用途向け集積回路）による機能、圧縮メモリ23と非圧縮

メモリ24はRAM206のメモリ領域、ビデオ出力部27はエンジンI/F207の機能をそれぞれ示している。

【0028】図6は、この発明によるデータ処理方法の第1の実施形態により1ページ分の印刷を行なう際の動作フロー図である。そこで、この図6のフローに沿って、図1を参照しながらページ印刷の動作を説明する。図2に示したパーソナルコンピュータ100内のプリンタドライバ101が、印刷すべき文書データをページ毎にポストスクリプト（以下、PSと記す）ファイルに変換してページプリンタ200へ送る。

【0029】ページプリンタ200がそのPSファイルを受信すると、図1に示したPSインタープリタ21が、それをプリンタエンジン208での通常印刷で用いられる解像度である600DPIで、ある大きさのブロック単位でラスターライズし、そのラスターライズしたデータに対して、圧縮部22がそのブロック単位で可変長可逆圧縮処理を試みる。このブロック単位については追って詳述する。

【0030】そして圧縮できれば、その結果の圧縮コードを圧縮メモリ23へストアする。この圧縮メモリ23は、300DPIで1ページ分に相当する容量を持つページメモリで、図5に示したRAM206のメモリ領域を使用する。圧縮できなかったら、ラスターライズされた600DPIのデータを非圧縮メモリ24へストアする。実際には、この非圧縮メモリ24は圧縮メモリ23を兼用することができるので、RAM206のメモリ容量を増加させる必要はない。

【0031】1ページ分の処理が終了したら、圧縮メモリ23へストアした圧縮画像をページの左上から順番に伸長部25で伸長し、600DPIの画像データに復元して、ビデオバッファを兼ねたビデオ出力部27へ送り、そこからシリアルなビデオ出力をプリンタエンジン208へ送出して印刷させる。非圧縮メモリ24にストアした600DPIの非圧縮画像データは、そのままビデオ出力部27へ送り、そのビデオ出力をプリンタエンジン208へ送る。

【0032】しかし、1ページ分の処理を終了する前に、圧縮メモリ23へストアする圧縮コード（圧縮メモリ+非圧縮メモリ）のデータ量が所定の大きさ、すなわち所定のメモリ容量を越えた場合は、「容量OK?」の判断でNOになる。その場合には、印刷すべき文書データのPSファイルを、PSインタープリタ21が上述と同じ600DPIの解像度でブロック単位でラスターライズし直し、そのラスターライズし直したデータに対して、ブロック単位で可変長可逆圧縮処理を施して、その圧縮コードを圧縮メモリ23にストアする。

【0033】そして、1ページ分の処理を終了すると、圧縮メモリ23へストアした圧縮画像をページの左上から順番に伸長部25で伸長し、600DPIの元のラス

ターライズした画像データに近似した画像データを再現してビデオ出力部27へ送り、そのビデオ出力を600DPIのプリンタエンジン208へ送出して印刷させる。なお、ここでいう「非可逆圧縮処理」とは、圧縮したデータを完全に復元することはできないが、ラスターライズされた元の画像データに近似した画像データを再現することは可能な圧縮処理であり、DCT(Discrete Cosine Transform)やBTC(Block Truncation Coding)などがある。

【0034】次に、上述したブロック単位の圧縮処理の詳細について図7乃至図14を参照して説明する。PSファイルをラスターライズするデータのブロック単位を、図7に示すようにドット(dot)のブロックワードBWとする。そして、図7の(a)又は(b)に示すように、このドット(dot)のブロックワードBWが主走査方向(a)あるいは副走査方向(b)にu個分連続して1ユニット(unit)を構成している。すなわち、1ユニット(unit)=u\*BWである。

【0035】そして、図5に示したCPU201は、図1のPSインタープリタ21としての機能によってPSファイルをラスターライズ(描画)した際、その1ユニットのデータが全て白ドットか否かを調べ、全て白の場合には圧縮部22を使用せずに直接圧縮メモリ23のホワイトマップテーブルWMTをNULL(2xffffffff)にする。黒ドットが1つでもあると、そのユニットを構成する各ブロックワードBW毎に、圧縮部22に圧縮処理を行なわせる。

【0036】図8はそのブロックワードBW及びユニットの具体例を示し、PSファイルをラスターライズしたデータに対して、64ドット(dot)の白ドット及び/又は黒ドットからなるブロックワードBWの8(line)分を1ユニットとする。すなわち、1ユニット(unit)=8\*BW=8\*64(dot)とする。

【0037】さらに、各ユニットに対して図9に示すように、32bitのホワイトマップテーブル(White Map Table: WMT)を用意する。そして、1ユニットのデータが全て白ドットなら、WMTをNULL(0xffffffff)とし、もし、黒ドットが1つ以上存在するなら、図10に示す32bitの圧縮データテーブル(Compression Data Table: CDT)の実アドレス(CDTアドレス)を格納する。

【0038】なお、CDTアドレスが示すレコードの長さは、図11に示すように8\*CDT=8\*32ビット(bit)=32バイト(byte)とする。CDTは1つのブロックワードBWに対応し、後述の圧縮方式を試みて、もし圧縮可能であったら、図10に示す圧縮データテーブルCDTのbit29~bit0に、その圧縮コードを格納する。また、もし圧縮不可能であったら、bit31=1とし、bit30~bit0に非圧縮時のデータ格納メモリアドレス(Ucompression Data Table Address: UD

Tアドレス)を格納する(図12)。

【0039】今回使用した圧縮方式は、次に示す2ステップからなる。まず始めに、対象となるブロックワードBWが全部白ドットか、全部黒ドットかを調べる。もし、そうであったらCDTアドレスをそれぞれ、0x00000000又は0x7fffffffとする。もし、そうでない場合には、先頭ドットが白ドットか黒ドットか(図9に示したCDTのbit30にて“0”か“1”で指定)に応じて、白ドットあるいは黒ドットのランレングスを図13に示すハフマン・コードで記述する。

【0040】例えば、図14の(a)に示す64ドットのブロックワードBWは、同図(b)に示す圧縮コードに変換される。なお、最後の白ドット(ランレングス=10)は、圧縮コードに含まれないが、最後の黒ドット(ランレングス=19)の後ろが全て白ドットである事で復元が可能である。また、CDTの余った領域には、1を書き込むようにする。

【0041】したがって、ラスターライズされたデータに対するこの可変長可逆圧縮処理を、図7に示したtドットのブロックワードBWが主走査方向又は副走査方向にu個並んだユニットを基本単位として行なう場合の処理は、次の各ステップによる。印刷すべきデータを1ページずつラスターライズするために必要なページバッファを前記ユニット単位に分割するステップ。その分割した各ユニットのデータが全白であるかどうかをチェックし、その結果をmビット(図9では32ビット)のホワイトマップテーブルに保存するステップ。

【0042】各ユニット毎にそのデータが全白でない場合に、ホワイトマップテーブルで指定されるアドレス領域にuブロック分のnビット(図10では32ビット)の圧縮データテーブル(CDT)を用意し、u個分のブロック毎のデータの圧縮処理結果を格納するステップ。各ブロック毎の圧縮処理が不可能な場合に、圧縮データテーブル(CDT)で指定されるアドレス領域にtビットの非圧縮データテーブル(UDT)を用意し、圧縮処理が不可能なブロックのデータをそのまま格納するステップ。

【0043】次に、この発明によるデータ処理方法の他の実施形態について説明する。図15は、前述のページプリンタ200の図5に示したコントローラ基板19及び600DPIのプリンタエンジン208による、ページ印刷の処理に係わる部分の以下に説明する各実施形態に共通の機能構成を示すブロック図である。この図5において前述した図1の構成と異なるのは、図1におけるビデオ出力部27に代えて、そのビデオ出力部の機能も兼ねた2値多値変換処理部26を設けた点のみである。

【0044】図16は、この発明によるデータ処理方法の第2の実施形態により1ページ分の印刷を行なう際の動作フロー図である。そこで、この図16のフローに沿って、図15を参照しながらページ印刷の動作を説明す

るが、前述の第1の実施形態と相違する部分についての説明する。

【0045】この図16に示す動作フローにおいて、図6の動作フローと異なるのは、図示は省略しているが、PSインタープリタ21がPSファイルのラスターライズを開始した時、その処理時間を計測するタイマをスタートさせる点と、1ページ分の処理を終了する前に、圧縮メモリ23へストアする圧縮コード(圧縮メモリ+非圧縮メモリ)のデータ量が所定のメモリ容量を越えた場合(「容量OK?」の判断でNOになった場合)の処理である。

【0046】その場合には、それまでに計測した処理時間が所定値より小さいか大きいかを判断する。そして小さかった場合は、図16のPSインタープリタ21によって、解像度を600DPI(第1の解像度)からそれより小さい300DPI(第2の解像度)に変更して、PSファイルを圧縮メモリ23へラスターライズする。計測した処理時間が所定値より大きかった場合は、先に600DPIの解像度でラスターライズされたデータを300DPI(第2の解像度)に変換するため、間引き処理を行なう。

【0047】その間引き処理は、例えば図19の(a)に示すような600DPIのデータの4×4ドット分を4分割して、2×2ドットずつの4つの部分とし、その各部分内の黒画素(斜線を施して示す)の数が、0~2のときはその部分全体を白画素とし、3~4のときはその部分全体を黒画素とするように変換すれば、(b)に示すように300DPIの2×2ドット分のデータに変換できる。

【0048】その後は、300DPIの解像度でPSファイルの残りのラスターライズを継続し、1ページ分の処理を終了したら、圧縮メモリ23に格納されている300DPIのラスターライズされた画像(ビットマップデータ)を左上から順番に、伸長部25を通さずに直接読み出して2値多値変換処理部26へ送る。そこで600DPIへの解像度変換を含む2値多値変換処理を行なって、そのビデオ出力を600DPIのプリンタエンジン208へ送出して印刷させる。その解像度変換を含む2値多値変換処理については、追って詳述する。

【0049】なお、この例において、圧縮メモリ23に1ページ分の圧縮コードを1ページ分格納できた場合には、前述の第1の実施の場合と同様に、圧縮メモリ23から圧縮コードを順次読み出し、伸長部25で伸長処理を行なって復元した600DPIのデータを2値多値変換処理部26へ送り込むが、その場合には解像度変換も2値多値変換処理も行なうことなく、図1におけるビデオ出力部27と同様に機能して、ビデオ出力をプリンタエンジン208へ送出する。

【0050】図17は、この発明によるデータ処理方法の第3の実施形態により1ページ分の印刷を行なう際の

動作フロー図である。そこで、この図17のフローに沿って、図15を参照しながらページ印刷の動作を説明するが、前述の第1の実施形態と相違する部分についてのみ説明する。

【0051】この図17に示す動作フローにおいて、図6の動作フローと異なるのは、印刷すべき文書データPSファイルを受信した後、そのページ言語中に含まれる印字範囲を示すデータにより、印字範囲が所定サイズ（所定の値）より小さいか大きいかを判断する点と、その判断結果が「大きい」の場合、および小さい場合（図6の場合と同様に、600DPIの解像度でラスターライズし、ブロック単位で可変長可逆圧縮処理を施し、その結果の圧縮コードを圧縮メモリ23に格納する）でも、1ページ分の処理が終了しないうちに圧縮コードのデータ量が圧縮メモリ23の容量を越えて、「容量OK?」の判断でNOになった場合の処理である。

【0052】これらの場合には、図17の動作フローに示すように、PSインタープリタ21が第1の解像度である600DPIより小さい300DPIの第2の解像度でPSファイルを圧縮メモリ23にラスターライズする。そして、1ページ分の処理を終了すると、図16の場合と同様に、圧縮メモリ23に格納されている300DPIのラスターライズされた画像（ビットマップデータ）を左上から順番に、伸長部25を通さずに直接読み出して2値多値変換処理部26へ送る。そこで600DPIへの解像度変換を含む2値多値変換処理を行なうと、そのビデオ出力を600DPIのプリンタエンジン208へ送出して印刷させる。

【0053】PSファイル中のページ言語に含まれる印字範囲を示すデータとは、例えば図20に示すように記述されており、これは図21に実線で示す用紙サイズに対して、印字範囲が破線で示す枠内であることを座標値で示している。

【0054】図18は、この発明によるデータ処理方法の第4の実施形態により1ページ分の印刷を行なう際の動作フロー図である。そこで、この図18のフローに沿って、図15を参照しながらページ印刷の動作を説明するが、前述の第3の実施形態と相違する部分についてのみ説明する。

【0055】この例を適用するページプリンタは、画像品質を優先するモードと処理速度を優先するモードとが選択可能なページプリンタである。そして、図18に示す動作フローにおいて、図17の動作フローと異なるのは、印刷すべき文書データPSファイルを受信した後、選択されている優先モードが「画像品質優先」か「処理速度優先」かを判断し、画像品質優先モードの場合には、600DPIの第1の解像度でラスターライズして可変長可逆圧縮処理を施すが、処理速度優先モードの場合には、上述の図17に示した例の場合と同様に300DPIの第2の解像度で直接圧縮メモリ23にラスターライ

ズする点だけである。

【0056】次に、図1及び図15に示した圧縮部22のハードウェア構成例を図22によって説明する。この圧縮部22は、リードブロック221、1ブロックワードBWのデータを格納できる容量である64bitのバッファ222、モディファイブロック223、及び圧縮ブロック224とからなる。

【0057】そして、図5に示したCPU201によるPSインタープリタの機能によって、ブロックワードBW単位で解像度600DPIでラスターライズされるデータを可変長可逆圧縮して、図5のRAM206を使用するページメモリである圧縮メモリ（WMT, CDT, UDTの各エリアを有する）23に、その圧縮コードをストアする。

【0058】リードブロック221は、CPU201によって算出された描画すべきユニット番号とそのブロックワードBWの位置の情報から、それに対応したホワイต์マップテーブルWMT、圧縮データテーブルCDT、及び非圧縮データテーブルUDTの圧縮データを圧縮メモリ23からリードし、圧縮前のブロックワードBWを64bitのバッファ222に再現するためのブロックである。

【0059】また、モディファイブロック223は、CPU201によって指定された新たに描画すべきブロックワードBWのデータ（Write Data）を、バッファ222にリード・モディファイ・ライト（読み出し・修正・書き込み）するためのブロックである。

【0060】そして、圧縮ブロック224は、リード・モディファイ・ライト処理されたバッファ222のデータを再圧縮するブロックで、その結果を圧縮メモリ23の先に読み出したユニット番号のBW位置に書き込む。すなわち、圧縮メモリ23に先に描画（ラスターライズ）されている圧縮データを、ブロックワード毎に新たに描画する画像の圧縮データに順次書き替えていく。

【0061】図23は、図5に示したCPU201による画像データの描画に係わる処理ルーチンのフローチャートである。CPU201は、まず以前の描画動作が終了し、現在図22に示したハードウェアが動作可能であるかをチェックし、もし動作可能であれば、描画すべきデータのユニット番号とブロックワードBWの位置を算出し、それをリードブロック221と圧縮ブロック224にセットする。

【0062】そして、モディファイブロック223に、そのブロックワードBWに新たに描画するデータ（Write Data）をセットすると、リードブロック221、モディファイブロック223、圧縮ブロック224の順にハードウェアが動作し、ページメモリである圧縮メモリ23への1ワードブロックBWの圧縮描画を実行し、1ページ分の描画終了まで上記の処理を繰り返す。

【0063】もし、CPUが図22に示したハードウェア



アの動作終了までアイドル状態である場合が多いシステムであれば、ユニット番号、BWの位置、および新たに描画すべきBWのデータを一時的に格納するために、CPUと圧縮部22の各ブロックとの間に、図22に仮想線で示すようにFIFO（先入れ先出し）メモリ225～227を設けるとよい。そして、その各FIFOメモリ225～227の残り容量がある大きさになったら、割り込みでCPUに知らせるように設計すれば、CPUは圧縮部22のハードウェアの状態を気にせずに、描画動作を実行することが可能になる。

【0064】次に、図15に示した2値多値変換処理部26による2値多値変換処理+解像度変換の処理について、図24～図33によって説明をする。図24は、図15に示した2値多値変換処理部26の機能構成を示すブロック図である。ページプリンタ200の図5に示したRAM206内における2値のページバッファ（図15における圧縮メモリ23に相当するページメモリ）とプリンタエンジン208との間（エンジンI/F207等）に2値多値変換処理部26を設け、そこで2値多値変換処理及び解像度変換の処理を行なう。

【0065】この2値多値変換処理部26では、RAM206のページバッファからの2値画像データに対して、その像域分離処理103、アウトライン補正処理104、及び疑似多値化処理・多値グレースケール処理105を並行して行ない、像域分離処理103によってグラフィック又は写真画像と認識された領域に対しては、疑似多値化処理・多値グレースケール処理105の処理結果を、それ以外の領域に対してはアウトライン補正処理104の処理結果を、それぞれ解像度変換を施した後に選択107し、プリンタエンジン208に対して出力する。

【0066】そこで、まず上記の像域分離処理103について説明する。この像域分離処理については、以下に示す制約を設ける。低線数のグラフィック、写真画像と文字、図形等で使用される網掛けパターンについては、その像域分離が事実上不可能であるため、高線数（約100線以上）のハーフトーン画像のみ認識する。ただし、この線数についての下限は変更することも可能である。

【0067】像域分離処理103の機能構成を図25に示し、以下その各機能を順番に説明する。

〔ハイライト・シャドウ部検出処理110と線数判定処理111〕中解像度のレーザプリンタの出力画像と云う限定された条件下では、文字及び線画像周辺には1ドットの孤立ドットは存在しない。また、低線数のグラフィック、写真画像領域に関しては、1ドットの孤立ドットはある領域内にはある限られた個数しか存在しない。そこで、ハイライト・シャドウ部検出処理110によって、孤立ドットを検出してグラフィック及び写真画像領域内のハイライト・シャドウ部を検出する。

【0068】そして、線数判定処理111により、その存在個数を計数することによって、高線数のグラフィック及び写真画像領域のみを抽出する。孤立ドットの検出には、例えば、図26の(a)、(b)に示す様な3×3画素のマッチングパターンによるマッチング処理を用いる。また、線数判定処理は、例えば、9×9画素内の孤立ドット個数 $S_x$ と閾値 $T_k$ により、 $S_x \geq T_k$ の条件が成立する画素を検出することによって行なう。そして、その検出された画素に所定の膨張処理を施し、最終結果Aとする。

【0069】なお、 $T_k$ は初期値が5で、数1の条件により変化する。この $T_k$ を変更することによって、線数判定の条件を変化させることが可能になる。例えば、原稿（用途）によって、この線数の判定条件を外部から指定できるようにすることも可能である。

【0070】

【数1】

```

if ( $S_x < T_k$ )      then  $T_k += 2$ ;
if ( $S_x = T_k$ )      then  $T_k += 1$ ;
if ( $S_x > T_k$ )      then  $T_k = T_k$ ;
if ( $S_x > T_k + 1$ )  then  $T_k -= 1$ ;
if ( $S_x > T_k + 2$ )  then  $T_k -= 2$ ;

```

【0071】〔中間濃度部検出処理112〕中間濃度部の写真画像には、ある大きさの白画素あるいは黒画素の塊が存在する。そして、線数が増える程その大きさが小さくなる。逆にいうと、ある大きさ以上の画素の塊を検出してそれを膨張させることによって、文字、線画、低線数のグラフィック、写真画像を検出することができる。図27に白画素及び黒画素の塊の例を示す。中間濃度部検出処理112は、このようなある大きさ以上の画素の塊を検出して、それを所定の大きさに膨張させる処理を行なって、最終結果Bとする。

【0072】〔領域検出処理113〕原画像に対し、領域検出処理113で所定の大きさの膨張処理を行なうことにより、文書内の文字、線画、写真、グラフィック画像の領域検出処理を行なう。この情報Cを基に、総合判定処理114を行なう。

【0073】〔総合判定処理114〕図25における線数判定処理111、中間濃度部検出処理112、及び領域検出処理113の出力結果A、B、Cに対して、Cが真“1”である画素の周辺領域に（ $A=1$ ）&（ $B=0$ ）が成立する画素が存在した場合に、その画素を高線数のグラフィック・写真画像とし、その画素を所定の大きさに膨張する。

【0074】以上が、像域分離処理103の処理内容であるが、この処理はディザ処理やハーフトーン・スクリーン処理で2値化された画像よりも、誤差拡散処理のような線数を持たない画像の方が認識率が高いことが実験で確かめられている。従って、図15における圧縮部22で圧縮できずに、PSインタープリタ21がPSファイルを300DPIで展開（ラスライズ）し直す時

に、イメージデータに関して誤差拡散処理で2値化するように処理する。また、誤差拡散処理はオリジナル画像の濃度値を精度よく保存するので、後述する疑似多値化処理にも適している。

【0075】次に、図24における疑似多値化処理・多値グレースケール処理105について説明する。例えば、図28に示すようなラプラシアン・フィルタによるエッジ強度に応じて、図29に示すようなアベレーシング・フィルタサイズを変化させ、2×2等の多値ディザ処理を施す。圧縮できずに300DPI展開された画像は、図30に示すような2×2の単純拡大法を適用して拡大した後に多値ディザ処理を施すことにより、画質劣化の少ない600DPI画像に復元することが可能になる。すなわち、文字線画以外の領域に対しては、平滑化フィルタによる疑似多値化処理で多値データに変換した後、第1の解像度(600DPI)に変換することになる。

【0076】最後に、図24におけるアウトライン補正処理104による解像度変換と、多値スムージング処理について説明する。300DPI展開された文字や線画像を600DPI画像に解像度変換する方法としては、既に実用化されているMOST技術などがある。ここでは、一例としてそのMOST技術による説明を図31～図33を参照して行なう。

【0077】まず、変換すべき着目画素すなわち図31の(a)に示す中心画素を、その隣接画素を参照して、同図の(b)に示すように12×12ドットの画素にスムージング拡大する。そして、その拡大率に応じて、等倍なら図32の(a)に示すように、その12×12ドット(そのうちの黒ドット数は42)に対して12×12の平滑化処理を施し、プリンタの多値レベルPに正規化する。すなわち、着目画素は次のようになる。

$$42 / (12 \times 12) \times P = 7 / 24 P$$

【0078】また、もし2×2倍の時には、図32の(b)に示すように、12×12ドットに拡大された画素を4つの6×6の画素に区切り(各区切り内の黒ドット数は、左上:0、右上:0、左下:12、右下:30)、その各々に対して6×6の平滑化処理を施す。従って、着目画素は、

$$0 / (6 \times 6) \times P = 0$$

$$0 / (6 \times 6) \times P = 0$$

$$12 / (6 \times 6) \times P = 1 / 3 P$$

$$30 / (6 \times 6) \times P = 5 / 6 P$$

の2×2ドットの多値濃度に変換される。

【0079】以下同様に、2×4倍の時には図32の(c)に示すよう8つの6×3ドットの画素に、3×3倍の時には(d)に示すように9つの4×4ドットの画素に、3×6倍の時には(e)に示すように18個の4×2ドットの画素にそれぞれ区切り、その各々に対して、6×3、4×4、4×2の各平滑化処理を施す。

【0080】図33は多値スムージング処理を行なうための内部ブロック図である。これは、着目画素に隣接する画素を蓄えるための入力ライン・バッファ部301と、その画素を拡大率に応じてスムージング拡大するためのTPM部302と、その結果を拡大率に応じて分割及び平滑化処理するための演算部303とで構成される。

【0081】この例では、上記処理の2×2倍のモードを使用することにより300DPIの文字や線画像を多値スムージング処理して、ジャギーを除去しながら600DPIに解像度変換する。すなわち、図24の像域分離処理103によって、文字線画領域と認識された領域に対しては、多値のスムージングを施して第1の解像度(600DPI)に変換する。

【0082】

【発明の効果】この発明によるページプリンタにおけるデータ処理方法を適用すれば、印刷すべき文書データをラスターライズしたデータを1ページ分格納するためのメモリの容量を大幅に削減でき、低価格で高解像度のページプリンタを提供することが可能になる。また、高解像度での圧縮データがメモリ容量を越えるような場合でも、印刷不能になるようなことなく、若干画質を落しても印刷可能にすることもできる。

【図面の簡単な説明】

【図1】図2乃至図4に示したページプリンタ200におけるPSファイルの文書データを印刷するための処理に係わる部分の第1の実施形態の機能構成を示すブロック図である。

【図2】この発明によるデータ処理方法を実施したページプリンタを用いた画像形成システムの構成例を示す図である。

【図3】図2におけるページプリンタの一例を示す外観図である。

【図4】同じくその内部機構の概略を示す縦断面図である。

【図5】図4におけるコントローラ基板19の内部ブロック図である。

【図6】この発明によるデータ処理方法の第1の実施形態により1ページ分の印刷行なう際の動作フロー図である。

【図7】PSファイルをラスターライズしたデータの1ユニット(unit)の構成を示す図である。

【図8】同じくそのブロックワードBWのドット数及び1ユニットのライン数の具体例を示す図である。

【図9】各ユニットに対するホワイトマップテーブル(WMT)のサイズを示す図である。

【図10】図9のWMTに格納する圧縮データテーブル(CDT)の構成を示す図である。

【図11】図10に示したCDTの実アドレス(CDTアドレス)のレコードの長さを示す図である。

【図12】非圧縮時のデータ格納メモリアドレス(UDTアドレス)をBWに格納する場合の説明図である。

【図13】白ドットあるいは黒ドットのランレングスを記述するハフマン・コードの説明図である。

【図14】圧縮コードの変換例を示す図である。

【図15】図2乃至図4に示したページプリンタ200におけるPSファイルの文書データを印刷するための処理に係わる部分の第2乃至第4の実施形態に共通の機能構成を示すブロック図である。

【図16】この発明によるデータ処理方法の第2の実施形態により1ページ分の印刷行なう際の動作フロー図である。

【図17】この発明によるデータ処理方法の第3の実施形態により1ページ分の印刷行なう際の動作フロー図である。

【図18】この発明によるデータ処理方法の第3の実施形態により1ページ分の印刷行なう際の動作フロー図である。

【図19】図16における600DPIから300DPIに解像度を変換するための間引き処理の例を示す説明図である。

【図20】図17における印刷範囲が所定サイズより大きい小さいかを判断するためのPSファイル中のページ言語に含まれる印字範囲を示すデータの例を示す図である。

【図21】同じくその印字範囲を示すデータによる印字範囲の説明図である。

【図22】図1及び図15に示した圧縮部22のハードウェア構成例を示すブロック図である。

【図23】図5に示したCPU201による画像データの描画に係わる処理ルーチンのフロー図である。

【図24】図15に示した2値多値変換処理部26の機能ブロック図である。

【図25】図24における像域分離処理の機能ブロック

図である。

【図26】図25における線数判定処理111で使用する3×3画素のマッチングパターンの例を示す図である。

【図27】図25における中間濃度部検出処理112で検出する白及び黒画素の塊の例を示す図である。

【図28】図24における疑似多値化処理・多値グレースケール処理105で使用するラプラシアン・フィルタの例を示す図である。

【図29】同じくアベレーシング・フィルタサイズの例を示す図である。

【図30】同じく2×2の単純拡大法の説明図である。

【図31】図24におけるアウトライン補正処理104による解像度変換のためのスムージング拡大の例を示す説明図である。

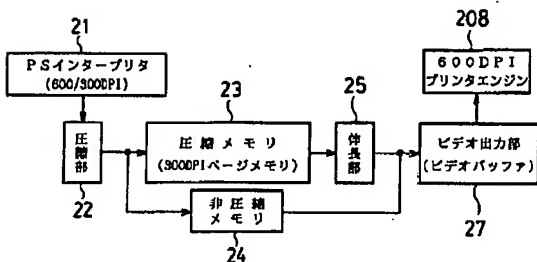
【図32】同じくその拡大率に応じた平滑化処理の説明に供する図である。

【図33】同じく多値スムージング処理を行なうための内部ブロック図である。

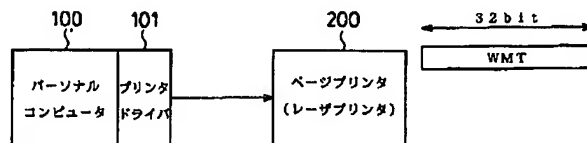
【符号の説明】

- 10: 感光体ドラム    11: 帯電部  
12: 光書込部    13: 現像部  
14: 転写部    15: 定着部  
19: コントローラ基板  
20: エンジンドライバ基板  
21: ホストスクリプト(PS)インタープリタ  
22: 圧縮部    23: 圧縮メモリ  
24: 非圧縮メモリ    25: 伸長部  
26: 2値多値変換処理部    27: ビデオ出力部  
100: パーソナルコンピュータ  
101: プリンタドライバ  
200: ページプリンタ    201: CPU  
206: RAM    208: プリンタエンジン

【図1】



【図2】

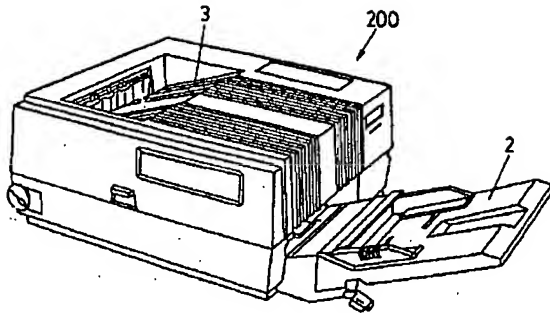


【図9】

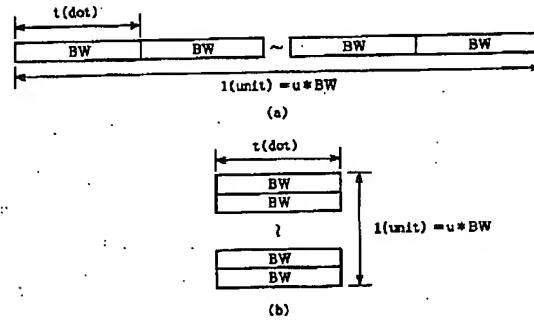
【図10】

bit No.	圧縮可能時	圧縮不可能時
31	0	1
30	0(White dot start) 1(Black dot start)	非圧縮時のデータ 格納メモリアドレス を示す
29	圧縮コードを格納する	
?		
0		

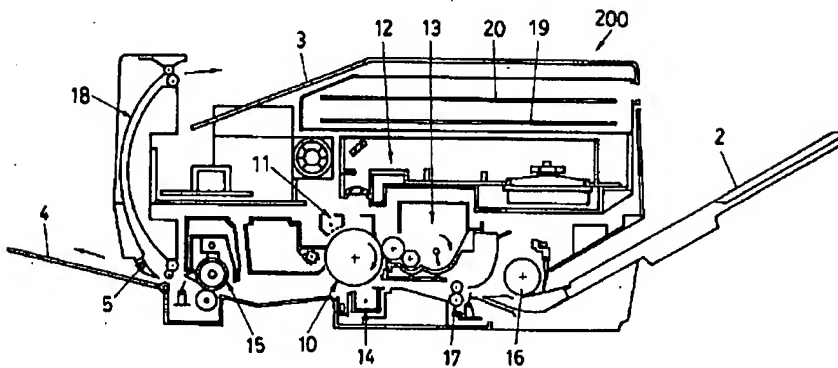
【図3】



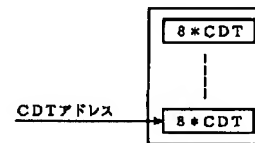
【図7】



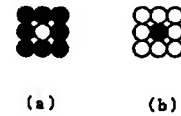
【図4】



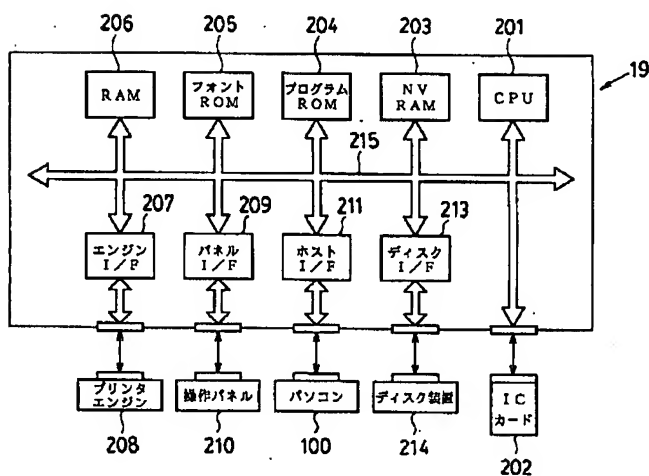
【図11】



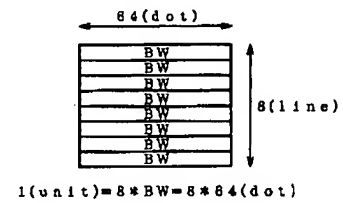
【図26】



【図5】

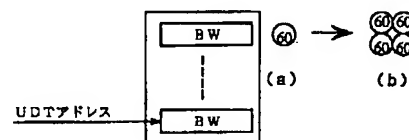


【図8】

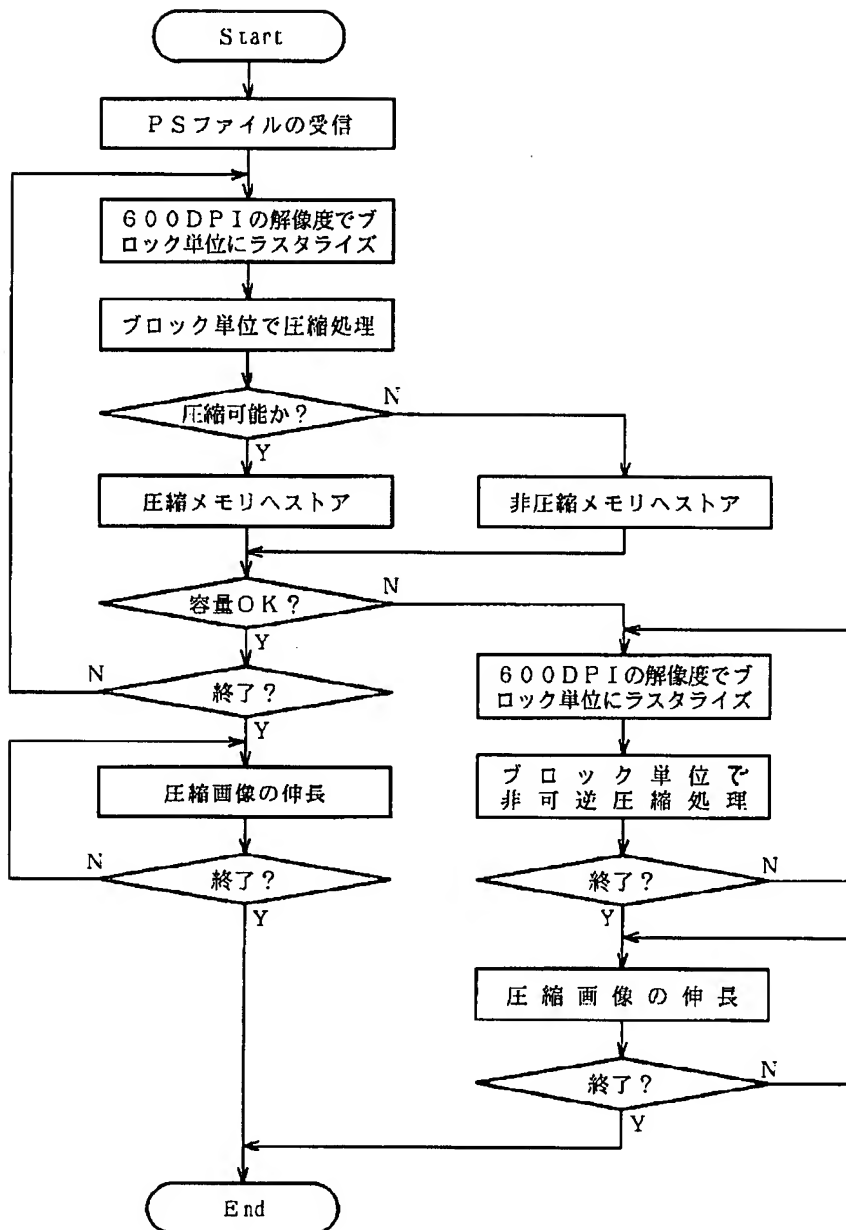


【図12】

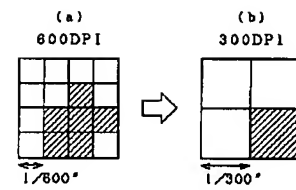
【図30】



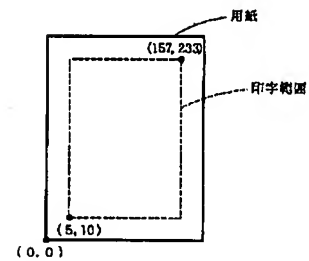
【図6】



【図19】



【図21】



【図29】

```

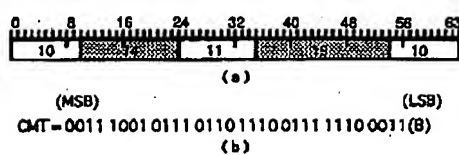
1 1 1 1 1
1 1 1 1 1
1 1 1 1 1
1 1 1 1 1
1 1 1 1 1

```

【図13】

value	#bits	code data
1	1	'0'
2,3	3	'10x'
4~7	5	'110xx'
8~15	7	'1110xxx'
16~31	9	'11110xxxx'
32~63	11	'111110xxxxx'
64~127	13	'1111110xxxxxx'

【図14】



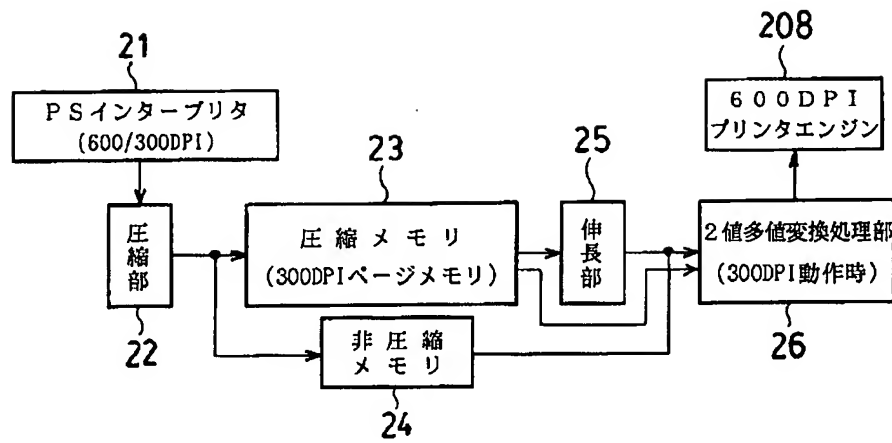
【図28】

```

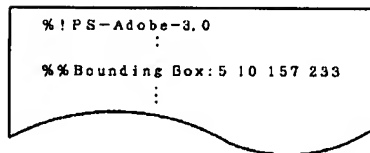
0-1 0
-1 4 -1
0-1 0

```

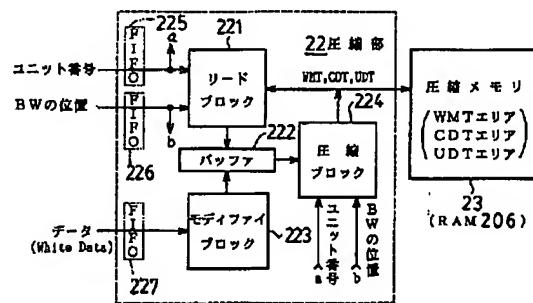
【図15】



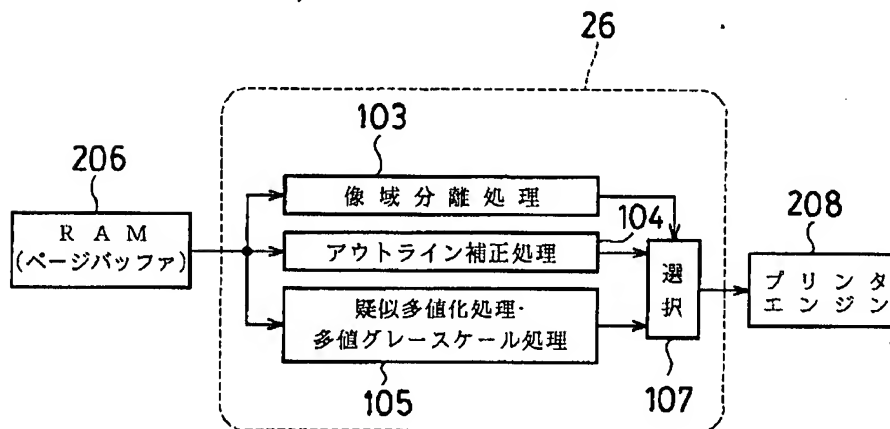
【図20】



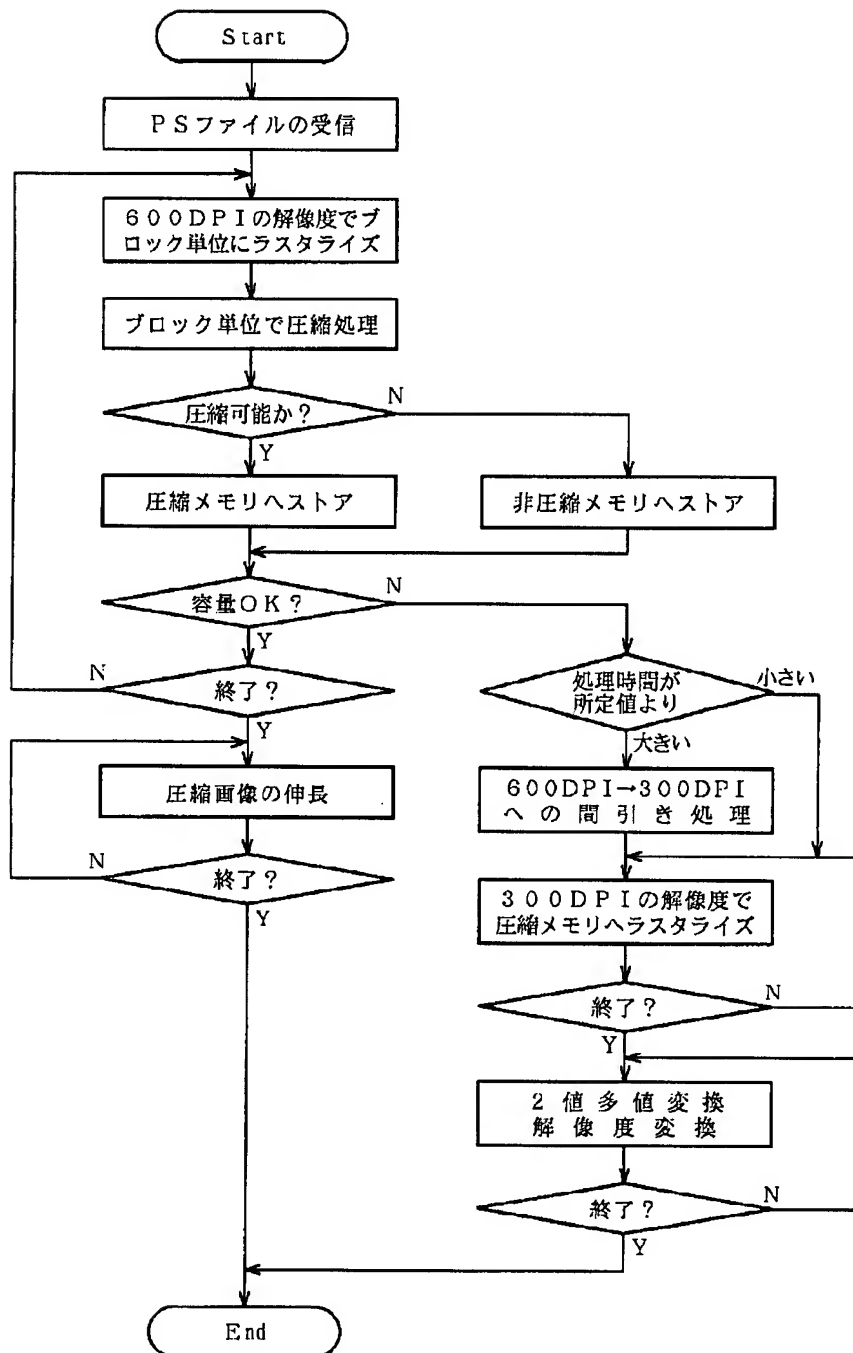
【図22】



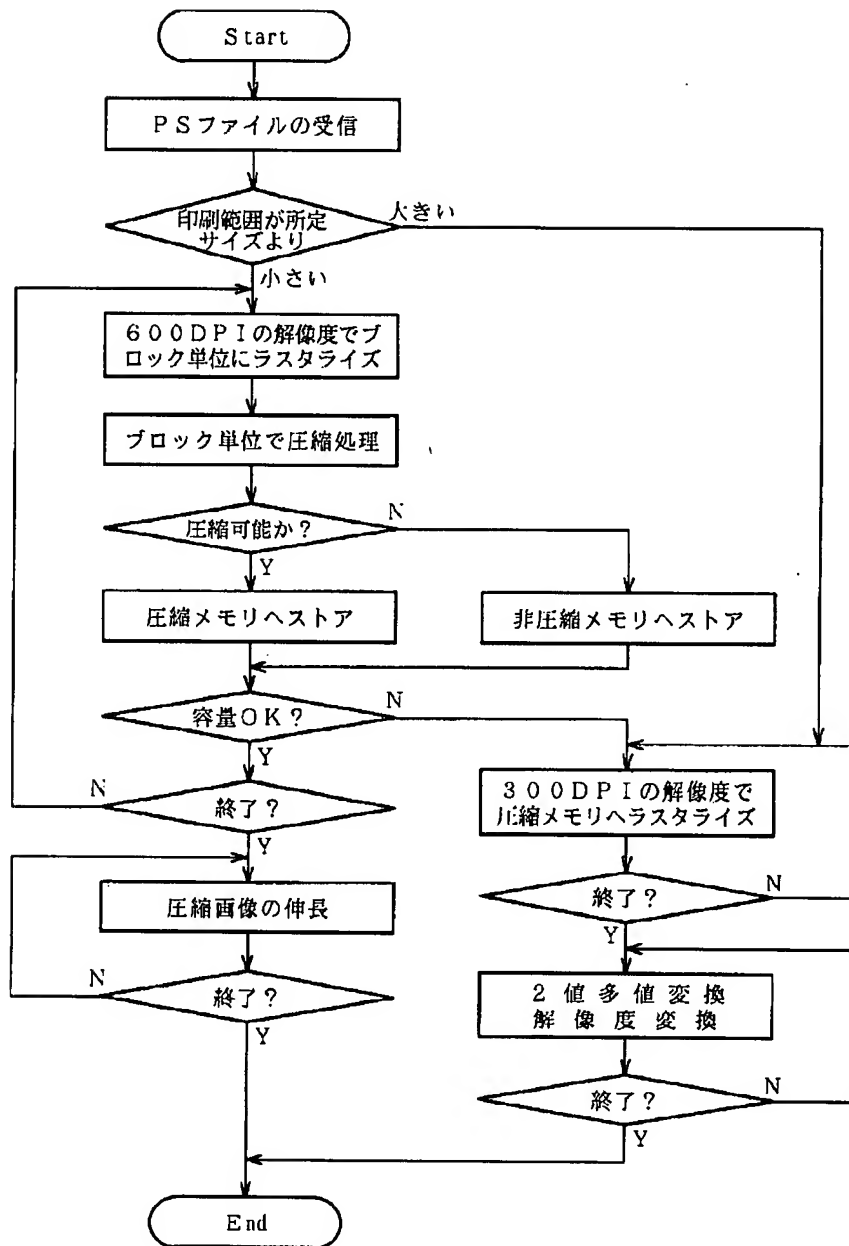
【図24】



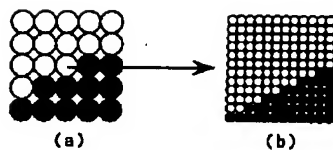
【図16】



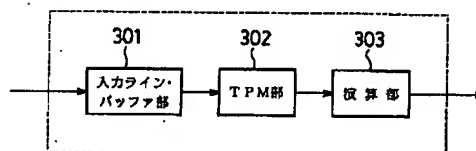
【図17】



【図31】

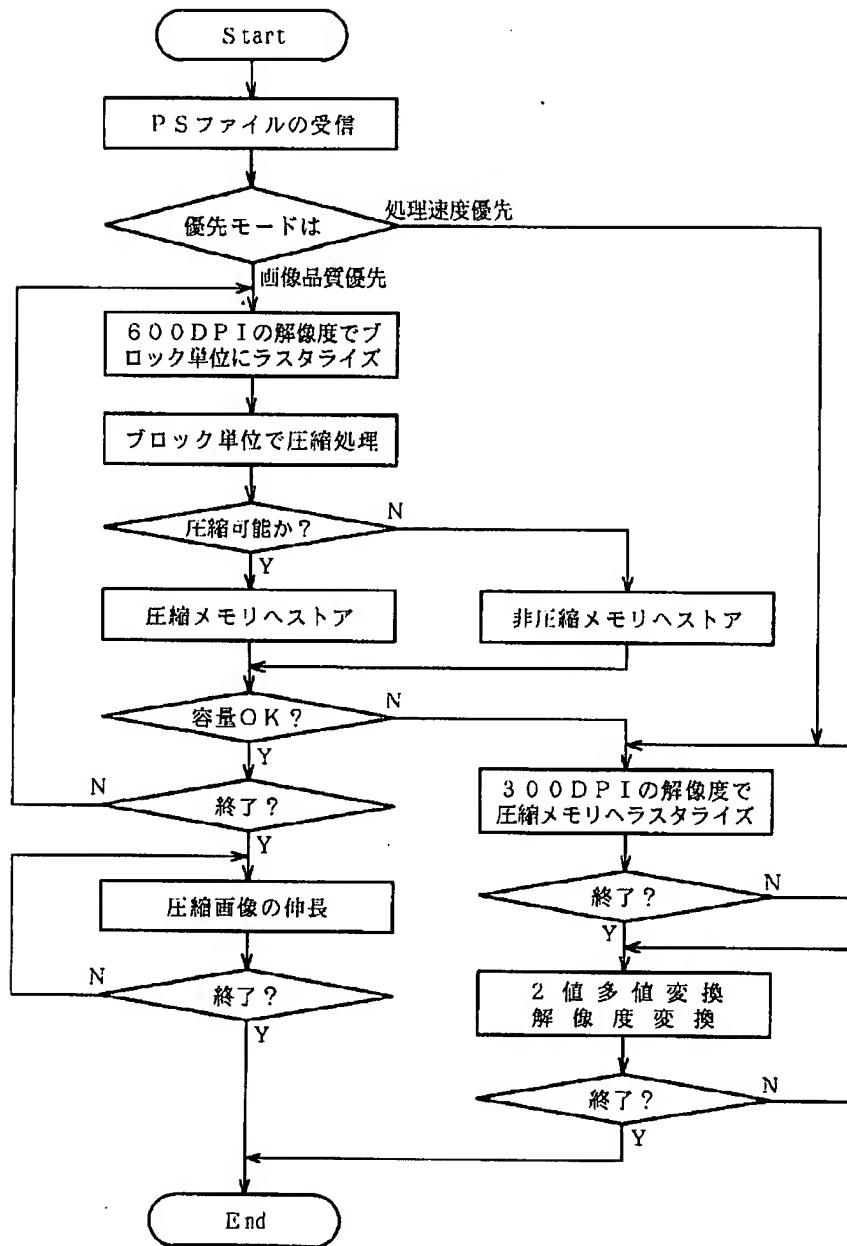


【図33】

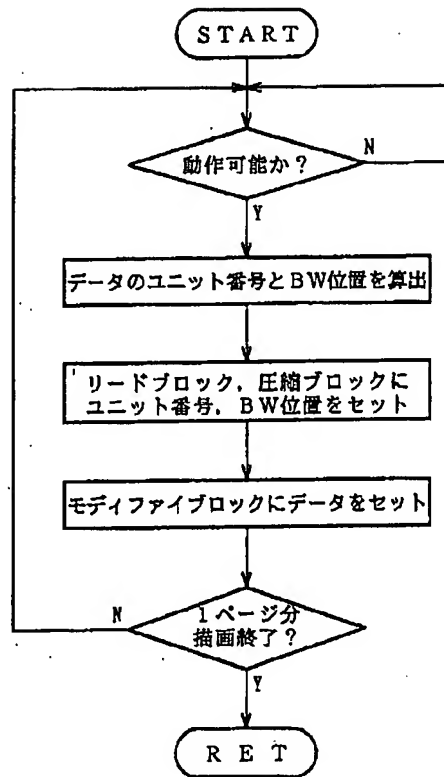




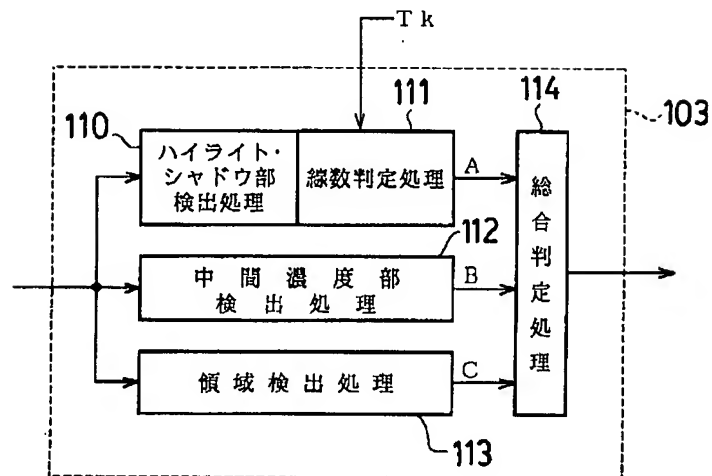
【図18】



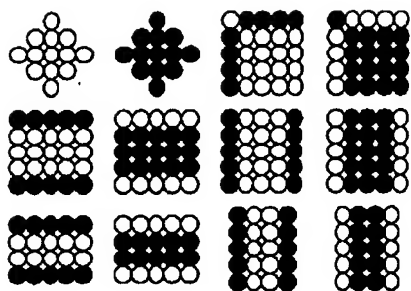
【図23】



【図25】



【図27】



【図32】

